PATENT ABSTRACTS OF JAPAN

(11)Publication number : 04-167134
(43)Date of publication of application : 15.06.1992

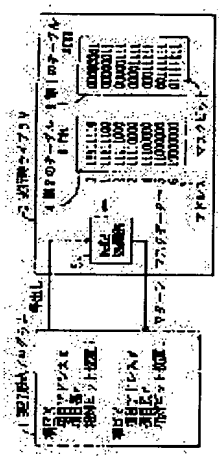
(51)Int.CI. G06F 9/305

(21)Application number : 02-294717
(22)Date of filing : 31.10.1990

(71)Applicant : FUJITSU LTD
(72)Inventor : KIMURA SHIGERU

(54) BIT TRANSCRIPTION SYSTEM

(57)Abstract:
PURPOSE: To transcribe the bits at a high speed by attaining a bit transcription system where the mask data is generated from a data table storing the least quantity of mask data and the transcribing and transcribed areas are cleared.
CONSTITUTION: A 1st table 3 consists of plural mask data of the processing unit width which contain the mask bits that mask the areas out of a cleared area and are continuous from the lower place side. A 2nd table 4 consists of plural mask data of the processing unit width which contain the mask bits continuous from the higher place side. A transcription processing part 5 defines the data obtained by applying an AND to a pair of mask data on one of tables 3 and 4 or a pair of mask data on both tables respectively as the 1st and 2nd mask data in accordance with the relative bit position and the bit length of a transcribing area. Then the part 5 clears the data on both tables and performs an AND between both cleared data. Thus the mask data are generated from the least quantity of data that contains the mask bits continuous from both boundaries. As a result, the bit transcribing speed is extremely increased.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's

decision of rejection]
[Date of extinction of right]

⑲日本国特許庁(JP)　⑪特許出願公開

⑫公開特許公報(A)　平4−167134

⑤1Int.Cl.[5]　識別記号　庁内整理番号　⑬公開　平成4年(1992)6月15日

G 06 F 9/305　9189−5B　G 06 F 9/30　340 A

審査請求　未請求　請求項の数 1　(全7頁)

⑤4発明の名称　ビット転記方式

⑳特　願　平2−294717

㉒出　願　平2(1990)10月31日

⑦2発明者　木村　茂　神奈川県川崎市中原区上小田中1015番地　富士通株式会社内

⑦1出願人　富士通株式会社　神奈川県川崎市中原区上小田中1015番地

⑦4代理人　弁理士　井桁　貞一

明細書

1. 発明の名称

ビット転記方式

2. 特許請求の範囲

処理単位の第1の領域のうちの相対ビット位置およびビット長で示される転記元領域をマスクして他の領域をクリアする第1のマスクデータと、処理単位の第2の領域のデータのうちの他の領域をマスクして転記先領域をクリアする第2のマスクデータとを生成し、第1および第2の領域の前記領域をクリアした後論理和して第1の領域から第2の領域の対応する該転記先領域にデータを転記するビット転記方式であって、

クリア領域外をマスクするマスクビットが下位側から連続する処理単位幅の複数のマスクデータから成る第1のテーブル(3)と、

該マスクビットが上位側から連続する処理単位幅の複数のマスクデータから成る第2のテーブル(4)と、

前記相対ビット位置およびビット長に応じ、第1および第2のテーブルのうちのいずれか一方のテーブルの1組のマスクデータ、または両方のテーブルの各1組のマスクデータを論理積したデータをそれぞれ第1のマスクデータとし、他方のテーブルの1組のマスクデータ、または両方のテーブルの各1組のマスクデータを論理和したデータをそれぞれ第2のマスクデータとして前記クリア処理を行う転記処理部(5)とを設け、

転記元領域の相対ビット位置およびビット長に基づき、下位側および上位側からマスクビットが連続する処理単位幅の複数のマスクデータにより第1および第2のマスクデータを生成して転記処理を行うことを特徴とするビット転記方式。

3. 発明の詳細な説明

〔概　要〕

本発明はビット転記方式に関し、

転記処理を高速化することを目的とし、

処理単位の第1の領域のうちの相対ビット位置

およびビット長で示される転記元領域をマスクして他の領域をクリアする第1のマスクデータと、処理単位の第2の領域のデータのうちの他の領域をマスクして転記先領域をクリアする第2のマスクデータとを生成し、第1および第2の領域の前記領域をクリアした後論理和して第1の領域から第2の領域の対応する該転記先領域にデータを転記するビット転記方式であって、クリア領域外をマスクするマスクビットが下位側から連続する処理単位幅の複数のマスクデータから成る第1のテーブルと、該マスクビットが上位側から連続する処理単位幅の複数のマスクデータから成る第2のテーブルと、前記相対ビット位置およびビット長に応じ、第1および第2のテーブルのうちのいずれか一方のテーブルの1組のマスクデータ、または両方のテーブルの各1組のマスクデータを論理積したデータをそれぞれ第1のマスクデータとし、他方のテーブルの1組のマスクデータ、または両方のテーブルの各1組のマスクデータを論理和したデータをそれぞれ第2のマスクデータとして前記クリア処理を行う転記処理部とを設け、転記元領域の相対ビット位置およびビット長に基づき、下位側および上位側からマスクビットが連続する処理単位幅の複数のマスクデータにより第1および第2のマスクデータを生成して転記処理を行うように構成する。

〔産業上の利用分野〕

本発明は、ＣＯＢＯＬ言語等で作成されたプログラムを実行する際、呼出されて転記処理を行う実行時ライブラリにおけるビット転記方式の改良に関する。

ＣＯＢＯＬ言語等では、ハードウエアレベルの情報であるビットを操作する言語仕様が強化されつつある。

ビットを操作する項目として、ハードウエアの処理単位（例えばバイト単位）で扱う外部ブール項目と、ビットそのものを扱う内部ブール項目とがあるが、項目間の転記処理は内部ブール項目の方がメモリが少なくてすむため、内部ブール項目として扱う方が主流となっている。

しかし、内部ブール項目の転記処理は外部ブール項目のそれよりも処理速度が遅く、より高速な転記処理方式が求められている。

〔従来の技術〕

第7図はプログラム構成図、第8図は内部ブール項目の転記処理説明図、第9図は従来の転記処理フローチャート図である。

ＣＯＢＯＬのソースプログラムで内部ブール項目Ｘから内部ブール項目Ｙへの転記は、

Ｘ　ＰＩＣ　１（２）
Ｙ　ＰＩＣ　１（２）
ＭＯＶＥ　Ｘ　ｔｏ　Ｙ

のように記述される。ここで（２）は項目長である。

このビット転記処理は、第7図に示すように、通常、実行時に呼出される実行時ライブラリ2で処理されており、コンパイラは、上記プログラムを、第8図に示すように、プロセッサの処理単位の領域（例えば1バイト）のアドレス（項目アドレス）ｘ，ｙ、項目長ｎ，相対ビット位置ｉ，ｊ（ここでは下位側からの先頭ビット位置）を設定し（第8図参照）、転記処理部10を呼び出す実行形式プログラム1 に翻訳する。

領域ｘ，ｙには他の項目が格納されている場合があり、呼出された転記処理部10は、指定された領域ｘ，ｙ内の他のデータを保存しつつ転記処理を行う。このため、

(1)　領域ｘの内容をワーク領域ｘ' に転記し、領域ｘ' のうち、相対ビット位置ｉ，項目長ｎで指定された領域を除く他の領域をクリアし、
(2)　転記先の相対ビット位置ｊに合わせ、
(3)　領域ｙのうち転記先領域のみをクリアし、
(3)　(2)と(3)の結果を論理和する

ことにより、領域ｘの転記対象データを領域ｙに転記する。

第9図は、以上の処理を実現する従来の転記処理例を示したものである。即ち、

①　ｘ領域の内容をワーク領域ｘ' に転記する。

② 指定された項目のうちの1ビットに着目し、他のビットをクリアする。⇒x'
③ 転記元と転送先の相対ビット位置i，jを比較してシフト要か否かをチェックする。
④ シフト要ならば、転記先の相対ビット位置jに合わせるため、領域x'内を1ビットづつシフトする。
⑤ マスクデータを生成する。これは、例えば、ワーク領域aに〔01〕（16進表示）を書込み、(i－j)ビット分左にシフトした後、1の補数をとる等の方法により生成する。
⑥ マスクデータと領域yと論理積する。⇒y
⑦ 領域x'と領域yのデータを論理和する。
⑦の結果をyに格納し、次のビットに着目して①～⑦の処理を繰り返す。
以上のごとく、転送元または転送先が2バイトにまたがる場合に対処できるよう、1ビットつづ転記処理を行っている。

〔発明が解決しようとする課題〕

以上のごとく、従来では、1ビットづつ転記処理を行っているため、処理時間がかかるという課題がある。

この1ビットづつ処理するのは、コンパイラが領域x，yとして、いずれか一方または双方が2バイトにまたがって指定する場合があり、これらを同一のアリゴリズムで対処できることと、複数ビットのマスクデータを生成するためには多くの処理ステップ数が必要であるという理由による。

本発明は、上記課題に鑑み、ビット転記処理を高速化するビット転記方式を提供することを目的とする。

〔課題を解決するための手段〕

第1図実施例の構成図および第3図本発明のマスクデータ生成説明図より対応する機能部分を抽出して説明する。

3は第1のテーブルで、クリア領域外をマスクするマスクビットが下位側から連続する処理単位幅の複数のマスクデータから構成される。

4は第2のテーブルで、マスクビットが上位側から連続する処理単位幅の複数のマスクデータから構成される。

5は転記処理部で、転記元領域の相対ビット位置およびビット長に応じ、第1のテーブル3および第2のテーブル4のうちのいずれか一方のテーブルの1組のマスクデータ、または両方のテーブルの各1組のマスクデータを論理積したデータをそれぞれ第1のマスクデータとし、他方のテーブルの1組のマスクデータ、または両方のテーブルの各1組のマスクデータを論理和したデータをそれぞれ第2のマスクデータとしてクリア処理を行い、クリア処理した両方のデータを論理和することにより転記する。

〔作　用〕

転記元領域と転記先領域のビット位置が対応する状態において、転記処理部5は、一方の境界からマスクビット"1"が連続する複数のマスクデータから成る第1のテーブル(BITR)3と、他方の境界からマスクビット"1"が連続する複数のマスクデータから成る第2のテーブル(BITL)4から、いずれか一方のテーブル、または両方のテーブルからマスクデータを読出し、組み合わせにより第1のマスクデータおよび第2のマスクデータを生成する。

転記元領域が第1の領域の上位側または下位側境界に接する場合、両境界とも接しない場合の3つのケースがあり、以下第3図を参照しつつ、説明する。

なお、処理単位幅を8ビットとして図示領域の右側を下位側、左側を上位側とし、第1のテーブルBITR3は下位側よりマスクビット"1"が連続するマスクデータ、第2のテーブルBITL4は上位側よりマスクビット"1"が連続するマスクデータをそれぞれ格納するものとし、またビット番号は下位側から昇順に0～7とする。またiは第1の領域（または第2の領域）における相対ビット位置（先頭ビット位置，左端を先頭、右端を最終とす

る）、ｎはビット長（項目長）である。

（ケースⅠ）　転記元領域の最終ビット位置ｋが第１の領域の右側境界に一致する場合（ｋ＝ｉ－ｎ＋１＝０）

第１のテーブルBITR3 から先頭ビット位置ｉとマスクビットの先頭位置ｉ'が一致するマスクデータ（第１図のテーブル配置ではアドレスｉのマスクデータ、BITR(i) と表す）を読出して第１のマスクデータとし、第２のテーブルBITL4 から（先頭ビット位置ｉ）＋１とマスクビットの最終位置ｋ'が一致するマスクデータ〔BITL(i) 〕を読出して第２のマスクデータとする。

（ケースⅡ）　転記元領域の先頭ビット位置ｉが第１の領域の左側境界と一致する場合（ｉ＝７）

第２のテーブルBITL4 から転送元領域の最終ビット位置ｋとマスクビットの最終位置ｋ'が一致するマスクデータ〔BITL(7-n) 〕を読出して第１のマスクデータとし、第１のテーブルBITR3 から（最終ビット位置ｋ）－１とマスクビットの先頭位置ｉ'が一致するマスクデータ〔BITR(7-n) 〕を読出して第２のマスクデータとする。

（ケースⅢ）　転記元領域がどちらの境界にも接しない場合

第２のテーブルBITL4 のうちの最終ビット位置ｋとマスクビットの最終位置ｋ'が一致するマスクデータ〔BITL(i-n) 〕と、第１のテーブルBITL3 のうちの先頭ビット位置ｉとマスクビットの先頭位置ｉ'が一致するマスクデータ〔BITR(i) 〕とを論理積したデータを第１のマスクデータとし、第１のテーブルBITR3 のうちの（最終ビット位置ｋ）＋１とマスクビットの先頭位置ｉ'が一致するマスクデータ〔BITR(i-n) 〕と、第２のテーブルBITL4 のうちの（先頭ビット位置ｉ）＋１とマスクビットの最終位置ｋ'が一致するマスクデータ〔BITL(i) 〕とを論理和したデータを第２のマスクデータとする。

そして、転記処理部3 は、第１のマスクデータと第１の領域のデータとを論理積し、第２のマスクデータと第２の領域のデータとを論理積してクリア処理を行った後、クリア処理した両データを論理和する。

以上のごとく、両境界からそれぞれマスクビットが連続する最小限のマスクデータにより第１および第２のマスクデータを生成するため、１ビットづつ転送処理を行う場合に比較して大幅に転記処理を高速化することが可能となる。

〔実施例〕

本発明の実施例を図を用いて詳細に説明する。

第１図は実施例の構成図、第３図は本発明のマスクデータ生成説明図、第２図は実施例の転記処理フローチャート図、第４図は処理Ａフローチャート図、第５図は処理Ｂフローチャート図、第６図は処理Ｃフローチャート図である。

以下、処理単位を８ビットとし、従来例と同一条件として説明する。

第１のマスクテーブルBITR3 は、下位側境界よりマスクビット"1" がそれぞれ１，２，・・７個連続し、それ以外のビットが"0" の７組のマスクデータから構成され、第１図のごとく配列されている。

第２のマスクテーブルBITL4 は、上位側境界よりマスクビット"1" がそれぞれ１～７個連続し、それ以外は"0" の７組のマスクデータから構成され、第１図のごとく配列されている。

転記処理部5 は、転記元および転記先の設定条件である項目アドレスｘ，ｙ，項目長ｎ，相対ビット位置ｉ、ｊが与えられて呼出されたとき、第１および第２のマスクテーブルBITR3 ，BTRL4 から転記処理に必要な第１および第２のマスクデータを以下に示す処理によって生成し、転記処理を行う。

第２図は、転記データの先頭ビットが上位側境界と一致する場合（ｉ＝７），転記データの最終ビットが下位側境界と一致する場合（ｉ－ｎ＋１＝０）およびいずれも一致しない場合を相対ビット位置ｉおよび項目長ｎにより判別してそれぞれの処理を行うように構成したものである。即ち、

(1)　転記元領域と転記先領域のビット位置が異なる（ｉ≠ｊ）か否かを判別し、異なる場合は処理

Dに進む。

(2) i＝jで、且つi＝7ならば転記元領域が上位側境界に一致するから処理Aに進む。

(3) i＝jで、且つj≠7ならば、最終ビット位置をチェックする。即ち、i－n＋1＝0ならば最終ビットが下位側境界と一致するから、処理Bに進む。

(4) i－n＋1≠0ならば、処理Cに進む。

以下処理A～処理Dの詳細を説明するが、領域xの他のビットを保存するため、すべての処理で領域xのデータをワーク領域x'（第1の領域に対応する）に転記して（処理A～Cのステップ①）領域x'から領域y（第2の領域に対応する）への転記処理を行う。

〔処理A〕第4図参照

前述の（ケースII）に相当するもので、（7－n）をアドレスとして第2のテーブルBITL4 および第1のテーブルBITR3 からマスクデータを読出し、それぞれ第1および第2のマスクデータとする。第4図は処理ステップを示したもので、

②（x'）・BITL(7-n) ⇒ x'

この論理積演算により、領域xのデータのうち転記元領域以外の領域がクリアされたデータが領域x'に得られる。

③（y）・BITR (7-n)⇒ y

これにより領域yのうち転記先領域のみクリアされたデータが領域yに得られる。

④（x'）＋（y）⇒ y

これにより、転記元領域のデータが領域yに転記される。

〔処理B〕第5図参照

前述の（ケースI）に相当する。（i）をアドレスとして第1のテーブルBITR3 および第2のテーブルBITL4 からマスクデータを読出し、処理Aと同様に以下の演算処理を行う。

②（x'）・BITR(i) ⇒ x'

③（y）・BITR (i)⇒ y

④（x'）＋（y）⇒ y

これにより、転記元領域のデータが領域yに転記される。

〔処理C〕第6図参照

（ケースIII）に相当するもので、第6図は第1および第2のマスクデータを生成する代わりに、第1および第2のテーブルから読出したそれぞれのマスクデータにより直接（x'），（y）に処理を施した例を示したものである。即ち、

〔BITR(i)・(x')〕・〔BITL(i-n)・(x')〕

を演算し、

〔BITL(i)・(y')〕＋〔BITR(i-n)・(y')〕

を演算し、両者の結果を論理和している。

〔処理D〕

i≠jの場合、図示省略したが、領域x'をシフトして転記先に合わせ、処理A～処理Cを適用する。なお、この場合は従来の処理を行ってもよい。

以上の処理で、項目が1処理単位の領域に限ったが、2領域にまたがる場合は、それぞれの領域で処理A～処理Cを適用すればよい。

以上のごとく、2組のテーブルよりマスクデータを生成してクリア処理を施し転記処理を行うため、項目長が複数ビットの場合でも一度の処理で行うことができ、また、マスクデータの生成が簡単になるため、転記処理の高速化が達成される。

〔発明の効果〕

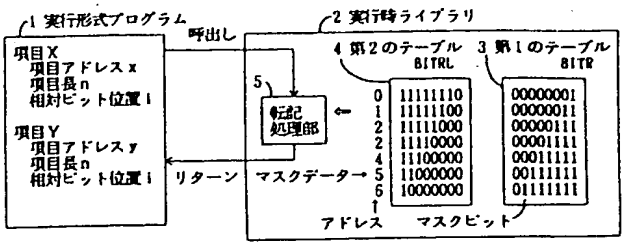
以上説明したように、本発明は、最小限のマスクデータを格納したテーブルからマスクデータを生成して、転記元および転記先領域のクリア処理を行うビット転記方式を提供するもので、ビット転記処理の高速化に多大な効果を奏する。

4. 図面の簡単な説明

第1図は実施例の構成図、第2図は実施例の転記処理フローチャート図、第3図は本発明のマスクデータ生成説明図、第4図は処理Aフローチャート図、第5図は処理Bフローチャート図、第6図は処理Cフローチャート図、第7図はプログラム構成図、第8図は内部プール項目の転記処理説明図、第9図は従来の転記処理フローチャート図である。

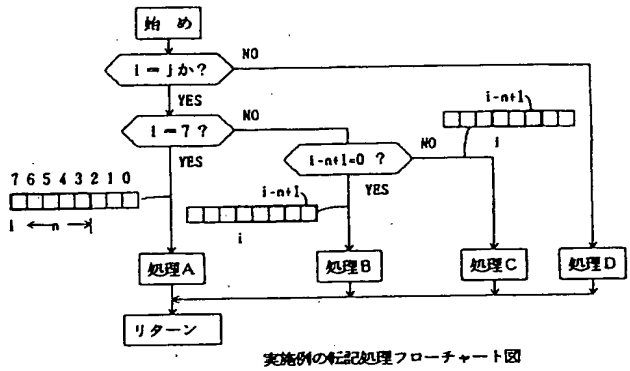
図中、1は実行形式プログラム、2は実行時ライブラリ、3は第1のテーブルBITR、4は第2のテーブルBITL、5，10は転記処理部である。

代理人　弁理士　井桁　貞一



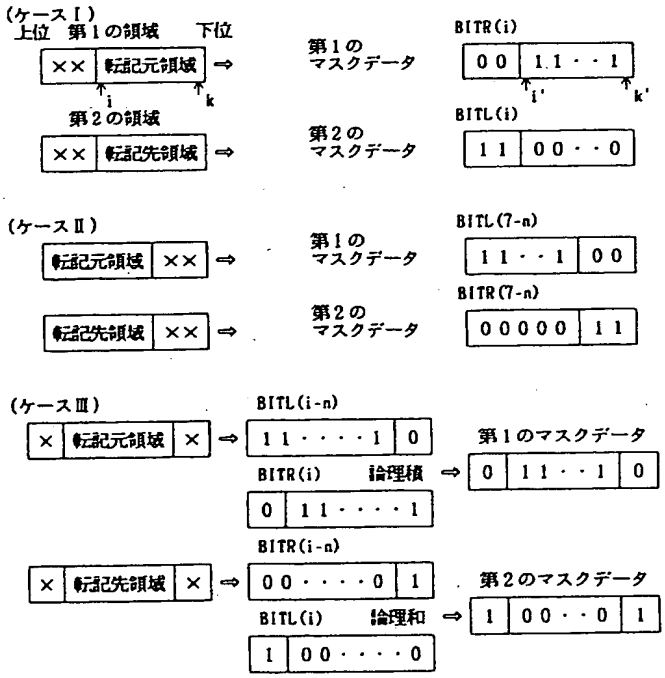
実施例の構成図

第　1　図



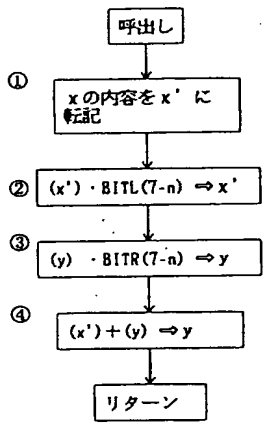
実施例の転記処理フローチャート図

第　2　図



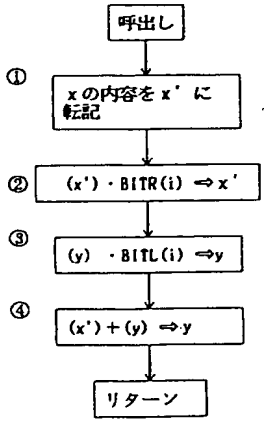
本発明のマスクデータ生成説明図

第　3　図



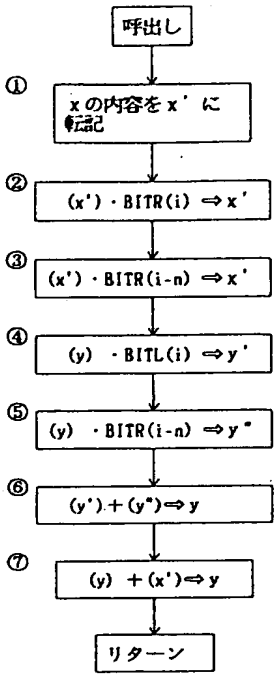
処理Aフローチャート図

第　4　図



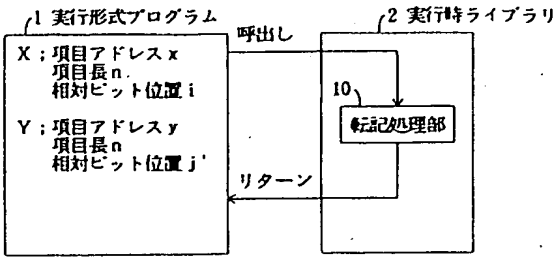
処理Bフローチャート図

第　5　図



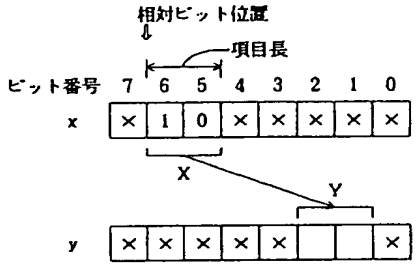
処理Cフローチャート図

第 6 図



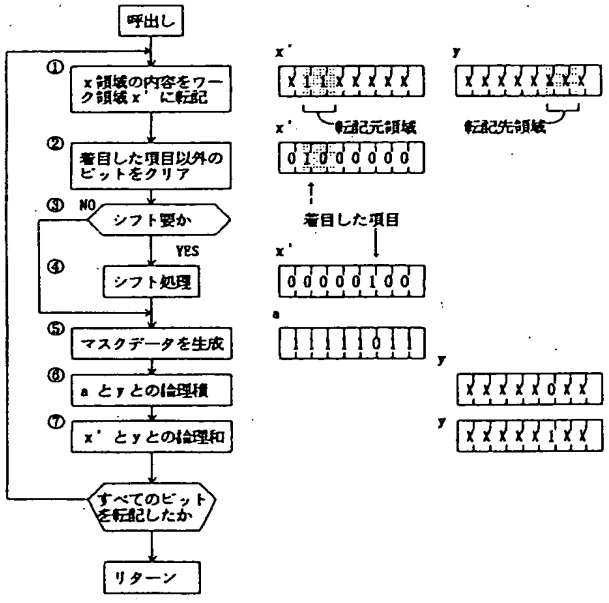
プログラム構成図

第 7 図



内部プール項目の転記処理説明図

第 8 図



従来の転記処理フローチャート図

第 9 図